CLAIMS

What is claimed is:

1. A system for efficiently performing memory intensive computations, the system comprising:

a data cache coupled to a first set of data and a second set of data and configured to perform a scan operation on at least a portion of the first set of data and an update operation on the second set of data with changes that have occurred in the first set of data;

an engine manager coupled to the data cache and configured to instruct the data cache to perform the scan and update operations; and

a solver coupled to the data cache and configured to perform computations on the second set of data.

2. The system of claim 1, further comprising an application specific plug-in coupled to the solver and configured to direct the solver to perform the computations on the second set of data.

3. The system of claim 1, wherein the system is configured to update the second set of data with substantially no more than the changes to the first set of data.

4. The system of claim 1, wherein the system is configured to update the second set of data with changes to the first set of data in a near-real-time fashion.

5. The system of claim 1, wherein the system is configured to update the second set of data with substantially no more than the changes to the first set of data that meet a given condition.

6. The system of claim 1, wherein the data cache coupled to a first set of data and a second set of data is coupled in a bidirectional fashion.

7. The system of claim 1, wherein the first set of data comprises metadata and application data.

8. The system of claim 1, wherein the solver comprises a generic algorithms module.

9. The system of claim 1, wherein the computations solve problems encountered in business applications.

10. A system for efficiently performing memory intensive computations, the system comprising:

a data cache having a second set of data, the data cache coupled to a first set of data and configured to perform a scan operation on at least a portion of the first set of data and an update operation on the second set of data with changes that have occurred in the first set of data;

an engine manager coupled to the data cache and configured to instruct the data cache to perform the scan and update operations; and

a solver coupled to the data cache and configured to perform computations on the second set of data.

11. The system of claim 10, further comprising an application specific plug-in coupled to the solver and configured to direct the solver to perform the computations on the second set of data.

12. The system of claim 10, wherein the system is configured to update the second set of data with substantially no more than the changes to the first set of data.

13. The system of claim 10, wherein the system is configured to update the second set of data with the changes to the first set of data in a near-real-time fashion.

14. The system of claim 10, wherein the system is configured to update the second set of data with substantially no more than the changes to the first set of data that meet a given condition.

15. The system of claim 10, wherein the data cache coupled to a first set of data is coupled in a bidirectional fashion.

16. The system of claim 10, wherein the first set of data comprises metadata and application data.

17. The system of claim 10, wherein the solver comprises a generic algorithms module.

18. The system of claim 10, wherein the computations solve problems encountered in business applications.

19. A system for efficiently performing memory intensive computations, the system comprising:

a database comprising a first set of data;

a plug-in configured to provide application specific functionality; and

an in-memory engine coupled to the database via a synchronization mechanism and comprising a second set of data, the in-memory engine configured to interface with the plug-in, the in-memory engine configured to perform computations on the second set of data to derive a first result, the in-memory engine configured to transfer, when the first set of data changes, these changes to the second set of data via the synchronization mechanism in order to update the second set of data and perform computations thereon to derive a second result.

20. The system of claim 19, wherein the system is configured such that the second set of data is updated with substantially no more than the changes to the first set of data.

21. The system of claim 19, wherein the system is configured to update the second set of data with the changes to the first set of data in a near-real-time fashion.

22. The system of claim 19, wherein the system is configured to update the second set of data with substantially no more than the changes to the first set of data that meet a given condition.

23. The system of claim 19, wherein the in-memory engine is coupled to the database in a bidirectional fashion.

24. The system of claim 19, wherein the database comprises metadata and application data.

25. The system of claim 19, wherein the in-memory engine comprises a data cache containing the second set of data.

26. The system of claim 19, wherein the in-memory engine comprises a generic algorithms module and an in-memory engine manager.

27. The system of claim 19, wherein the computations solve problems encountered in business applications.

28. A method of efficiently performing memory intensive computations, the method comprising:

performing, using a data cache, a scan operation on at least a portion of a first set of data;

performing, using the data cache, an update operation on a second set of data with changes that have occurred in the first set of data; and

performing computations, using a solver, on the second set of data.

29. The method of claim 28, wherein the second set of data is created from substantially all of the first set of data.

30. The method of claim 28, wherein the second set of data is created from all of the first set of data.

31. The method of claim 28, wherein the second set of data is updated with substantially no more than the changes to the first set of data.

32. The method of claim 28, wherein the second set of data is updated with the changes to the first set of data in a near-real-time fashion.

33. The method of claim 28, wherein the second set of data is updated with substantially no more than the changes to the first set of data that meet a given condition.

34. The method of claim 28, further comprising updating the first set of data with the second set of data.

35. The method of claim 28, wherein the computations solve problems encountered in business applications.

36. A system for efficiently performing memory intensive computations, the system comprising:

scanning means for performing, using a data cache, a scan operation on at least a portion of a first set of data;

update means for performing, using the data cache, an update operation on a second set of data with changes that have occurred in the first set of data; and

solving means for performing computations, using a solver, on the second set of data.

37. A computer-readable medium for efficiently performing memory intensive computations, the computer-readable medium comprising:

instructions for performing, using a data cache, a scan operation on at least a portion of a first set of data;

instructions for performing, using the data cache, an update operation on a second set of data with changes that have occurred in the first set of data; and

instructions for performing computations, using a solver, on the second set of data.